In the claims:

All of the claims standing for examination are presented below. There are no amendments to the claims or specification made in the present response.

What is claimed is:

B1

1. (Currently amended) In a ~~multi-streaming~~ processor having multiple processor streams supporting multiple data threads, a system for fetching instructions from individual ones of the multiple streams to a pipeline, comprising:

a fetch algorithm for selecting from which stream to fetch instructions; and

a branch predictor for forecasting whether a branch alternative of a branch instructions will be taken;

wherein the prediction by the branch predictor is used by the fetch algorithm in determining from which stream to fetch.

2. (Original) The system of claim 1 wherein a prediction that a branch will not be taken precipitates no change in the fetching process.

3. (Original) The system of claim 1 wherein a prediction that a branch will be taken results in switching fetching to a different stream if no target address is provided by the predictor.

4. (Original) The system of claim 1 wherein the branch predictor determines a probability that a branch alternative will be taken, and the probability is used by the fetch algorithm in determining from where to fetch next instructions.

5. (Original) The system of claim 1 wherein the forecast of the branch predictor is also used by a dispatch algorithm in selecting instructions from the pipeline to dispatch to functional units.

B

B1

6. (Currently amended) In a ~~multi-streaming~~ processor having multiple processor streams supporting multiple data threads, a system for fetching instructions from individual ones of the multiple streams to a pipeline, comprising:

a fetch algorithm for selecting from which stream to fetch instructions; and

one or both of a branch predictor for forecasting whether a branch alternative of a branch instructions will be taken, or a hit-miss predictor for forecasting whether instructions will hit or miss a data cache;

wherein the prediction by either or both of the predictors is used by the fetch algorithm in determining from which stream to fetch.

7. (Original) The system of claim 6 wherein a prediction that a branch will not be taken or that an instruction will hit the data cache precipitates no change in the fetching process.

8. (Original) The system of claim 6 wherein a prediction that a branch will be taken or that an instruction will miss a data cache results in switching fetching to a different stream if no target address is provided by the predictor.

9. (Original) The system of claim 6 wherein one or both of the branch predictors determine a probability that a branch alternative will be taken or that an instruction will miss the cache, and the probability is used by the fetch algorithm in determining from where to fetch next instructions.

10. (Original) The system of claim 6 wherein the forecast of one or both predictors is also used by a dispatch algorithm in selecting instructions from the pipeline to dispatch to functional units.

11. (Currently amended) A ~~multi-streaming~~ processor having multiple processor streams supporting multiple data threads comprising:

B

a fetch algorithm for selecting from which stream to fetch instructions; and

a branch predictor for predicting whether jumps proposed by branch instructions will be taken or not;

wherein a prediction by the branch predictor is used by the fetch algorithm in determining from which stream to fetch.

12. (Original) The processor of claim 11 wherein a prediction that a branch will not be taken precipitates no change in the fetching process.

13. (Original) The processor of claim 11 wherein a prediction that a branch will be taken results in switching fetching to a different stream if no target address is provided by the predictor.

14. (Original) The processor of claim 11 wherein the branch predictor determines a probability for whether a branch will be taken, and the probability is used by the fetch algorithm in determining from where to fetch next instructions.

15. (Original) The processor of claim 11 wherein the forecast of the branch predictor is also used by a dispatch algorithm in selecting instructions from the pipeline to dispatch to functional units.

16. (Currently amended) A ~~multi-streaming~~ processor having multiple processor streams supporting multiple data threads, comprising:

multiple physical hardware streams for running individual software data threads;

a data cache;

a fetch algorithm for selecting from which stream to fetch instructions; and

one or both of a branch predictor for forecasting whether a branch alternative of a branch instructions will be taken, or a hit-miss predictor for forecasting whether instructions will hit or miss a data cache;

wherein the prediction by either or both of the predictors is used by the fetch algorithm in determining from which stream to fetch.

B1

17. (Original) The processor of claim 16 wherein a prediction that a branch will not be taken or that an instruction will hit the data cache precipitates no change in the fetching process.

18. (Original) The processor of claim 16 wherein a prediction that a branch will be taken or that an instruction will miss a data cache results in switching fetching to a different stream if no target address is provided by the predictor.

19. (Original) The processor of claim 16 wherein one or both of the branch predictors determine a probability that a branch alternative will be taken or that an instruction will miss the cache, and the probability is used by the fetch algorithm in determining from where to fetch next instructions.

20. (Original) The processor of claim 16 wherein the forecast of one or both predictors is also used by a dispatch algorithm in selecting instructions from the pipeline to dispatch to functional units.

21. (Currently amended) In a ~~multi-streaming~~ processor having multiple processor streams supporting multiple data threads, a method for fetching instructions from individual ones of multiple streams as instruction sources to a pipeline, comprising the steps of:

(a) on loading a branch instruction, making a prediction by a branch predictor as to whether a branch will be taken or not; and

(b) if the prediction is that the branch will be taken, altering the source of the fetch if no target address is provided by the predictor.

22. (Original) The method of claim 21 wherein the predictor determines a

B

probability, and the probability is used in determining fetch source.

B1

23. (Original) In a ~~multi-streaming~~ processor having multiple processor streams supporting multiple data threads and a data cache, a method for fetching instructions from individual ones of multiple streams as instruction sources to a pipeline, comprising the steps of:

(a) on loading an instruction, making a prediction by one or both of a branch predictor as to whether a branch will be taken if the instruction is a branch instruction, or by a hit-miss predictor as to whether the instruction will hit the data cache; and

(b) discriminating from which stream to continue to fetch according to prediction made.

24. (Original) The method of claim 23 wherein the predictor or predictors determine a probability, and the probability is used in determining fetch source.

B